Advances in Sensitivity-based Nonlinear Model Predictive Control and Dynamic Real-time Optimization

L. T. Biegler^a, X. Yang^a, G. G. Fischer^b

^a*Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA*

^b*Petroleo Brasileiro S.A, Refineria Henrique Lage, REVAP, Sao Jose dos Campos, SP, BRASIL*

Abstract

Recent results in the development of efficient large-scale nonlinear programming (NLP) algorithms have led to fast, on-line realizations of optimization-based methods for nonlinear model predictive control (NMPC) and dynamic real-time optimization (D-RTO), with predictive nonlinear dynamic (e.g., first principle) models. For NMPC, optimization-based controllers are developed that lead to well-understood stability and robustness properties, even for large, complex plant models. The realization of NMPC requires the application of a fast NLP solver for time-critical, on-line optimization, as well as efficient NLP sensitivity tools that require 2-3 orders of magnitude less computation than the NLP solution. This leads to advanced step NMPC (asNMPC), which essentially eliminate computational delay. We also extend these capabilities to Dynamic Real-time Optimization (D-RTO) with more general stage costs that are economically based. This overview also extends input to state stability (ISS) properties for asNMPC to handle active set changes, and also for D-RTO through convex regularizations. Two large scale distillation case studies, based on nonlinear first principle models, are presented that demonstrate the effectiveness of these approaches.

Keywords: Nonlinear Programming, Nonlinear Model Predictive Control, Dynamic Real-time Optimization

*Corresponding author

Email address: biegler@cmu.edu (L. T. Biegler)

Preprint submitted to Elsevier

January 27, 2015

1. Introduction

For over three decades, Real-Time Optimization (RTO) and Model Predictive Control (MPC) have emerged as essential technologies for optimal process operation in the chemical and refining industry. More recently, MPC has been extended to Nonlinear Model Predictive Control (NMPC) in order to realize high-performance control of highly nonlinear processes. Moreover, for many applications there is a need for RTO to evolve from steady-state optimization models to dynamic models, especially for systems, such as batch and cyclic processes, that are never in steady state. Both NMPC and dynamic real-time optimization (D-RTO) allow the incorporation of first principle process models, which lead to on-line optimization strategies consistent with higher-level tasks, including scheduling and planning. The realization of both of these tasks requires fast optimization algorithms. A major concern is that computational times needed to solve these large-scale optimizations lead to feedback delays in implementation that can degrade performance and possibly destabilize the process. In addition, state estimation of the process must be accomplished in a similar, efficient manner. Nonlinear model predictive control for tracking and so-called “economic” stage costs, as well as associated state estimation tasks, are reviewed, formulated and analyzed in considerable detail in [46, 36]. Due to advances described in [8, 35], fundamental stability and robustness properties of NMPC are well-known, and many of the key issues related to the applicability and relevance of NMPC are understood. Moreover, the availability of detailed dynamic process models for off-line process analysis and optimization allows NMPC to be realized on challenging process applications. Nevertheless, an important hurdle is the cost and reliability of on-line computation; lengthy and unreliable optimization calculations lead to unsuccessful controller performance.

Several advances to NMPC address the important problem of computational delay. Newton-type strategies for constrained nonlinear processes were originally proposed in [32]; here the nonlinear dynamic model is linearized around a nominal trajectory, and a quadratic program (QP) is solved at every sampling time. More recently, a real-time iteration NMPC was proposed in [11] where only one Newton or QP iteration of the NLP is executed on-line at every sampling time, instead of solving the NLP completely. More generally, NMPC strategies have been developed that separate optimization problem into an off-line NLP based on predicted states, and fast on-line calculation for the actual state [11, 57, 60, 54, 53, 39, 34]. In all of these cases, the dynamic model is optimized over a moving horizon and sensitivity-based updates are made on-line.

The ability to perform nonlinear model-based control extends naturally to dynamic real-time optimization (D-RTO). Current practice in process applications decomposes economic optimization into two layers. First, real-time optimization (RTO) optimizes an economic objective with steady state models, leading to a setpoint handled by the lower-level control layer. The advanced control layer (using, e.g., NMPC) then tracks the setpoint to achieve a new steady state. However, this two-layer approach assumes that model disturbances and transients are neglected in the RTO layer [13]. Moreover, model inconsistency between layers and unresolved transient behavior may lead to unreachable setpoints [48]. Also, since the control layer has no information on dynamic economic performance, it may generate trajectories that simply track suboptimal setpoints to steady state [47, 25].

Recent studies on dynamic real-time optimization (D-RTO) have reported significant performance improvements with economically-oriented NMPC formulations [59, 47, 13, 2]. In addition, stability theory supporting economically-oriented NMPC requires development beyond the mature results for setpoint tracking based on a discrete Lyapunov analysis. This problem formulation and stability analysis must be modified to ensure a stable and robust D-RTO implementation, especially if optimum steady state operation is expected.

Finally, knowledge of the plant state is essential for realization of NMPC and D-RTO. In practice state information can only be inferred through a set of noisy measurements, in combination with the dynamic process model. For linear systems this is done via Kalman Filters (KF). Based on linearizations of the nonlinear plant model, Extended Kalman Filters (EKF) have typically been applied [26, 7]. Moreover, when integrating EKF and NMPC, a local separation principle has been established between the estimation and control problem [23, 24]. On the other hand, EKF may have poor performance for highly nonlinear systems [9, 42], thus spawning related estimation methods that include the Unscented Kalman Filter [28], the Ensemble Kalman Filter [14], and the Particle Filter [4]. While none of these methods deal with bounds on the states, recent extensions [51, 50, 43, 42], have been developed to address these features.

On the other hand, the state estimation problem can also be formulated directly as an NLP through Moving Horizon Estimation (MHE), which uses a moving window of past measurements to find the optimal state estimates with an objective function based on maximum likelihood concepts. MHE has very desirable asymptotic stability properties [45] with bounds on plant states handled directly by the NLP solver. Efficient algorithms for MHE are presented in [49, 60, 58, 1, 33, 37], which also address computational delay.

A comprehensive survey of sensitivity-based concepts for MHE, NMPC and

D-RTO was presented in [5]. There we addressed recent results for NMPC, MHE and D-RTO that are based on *advanced step* concepts that particularly focus on efficient NLP algorithms for background solutions, along with on-line updates based on NLP sensitivity. This leads to asNMPC and asMHE, respectively. Moreover, recent MHE and asMHE extensions to efficient updating of arrival costs and outlier detection were developed and demonstrated on a distillation case study.

This study builds on the survey in [5] and also focuses on new results related to sensitivity-based NMPC and D-RTO. In the next section we present the basic NMPC problem formulation and review an optimization framework based on interior-point NLP solvers and sensitivity concepts. Section 3 presents advanced step NMPC (asNMPC) strategies and related stability properties. New results for ISS stability that incorporate controller clipping to deal with active set changes are also presented. Section 4 then describes a multi-step extension of asNMPC that allows very large process models to be solved in background over multiple time steps. This approach is illustrated on a large-scale distillation example. Section 5 then discusses recent updates for economic NMPC properties and demonstrates their impact with a D-RTO distillation case study. Finally, Section 6 summarizes the paper along with directions for future work.

2. NLP Strategies for NMPC

We begin with the following discrete-time nonlinear dynamic model of the plant with uncertainties:

$$\begin{aligned} x(k+1) &= \hat{f}(x(k), u(k), w(k)) \\ &= f(x(k), u(k)) + g(x(k), u(k), w(k)) \end{aligned} \quad (1)$$

where $x(k) \in \mathfrak{R}^{n_x}$, $u(k) \in \mathfrak{R}^{n_u}$ and $w(k) \in \mathfrak{R}^{n_w}$ are the plant states, controls and disturbance signals, respectively, defined at time steps t_k with integers $k > 0$. The mapping $f : \mathfrak{R}^{n_x+n_u} \mapsto \mathfrak{R}^{n_x}$ with $f(0,0) = 0$ represents the nominal model, while the term $g : \mathfrak{R}^{n_x+n_u+n_w} \mapsto \mathfrak{R}^{n_x}$ is used to describe modeling errors, estimation errors and disturbances. $f(\cdot, \cdot)$ and $g(\cdot, \cdot, \cdot)$ are continuous. We assume that the noise $w(k)$ is drawn from a bounded set $\mathcal{W}$.

With this model description, we compute an estimate of the current state $x(k)$ that can be used for our model-based controller, defined by the following NMPC

problem:

$$J_N(\eta_0) := \min_{z_l, v_l} \quad \Psi(z_N) + \sum_{l=0}^{N-1} \psi(z_l, v_l) \quad (2a)$$

$$\text{s.t. } z_{l+1} = f(z_l, v_l) \quad l = 0, \dots, N-1 \quad (2b)$$

$$z_0 = \eta_0 \quad (2c)$$

$$z_l \in \mathbb{X}, v_l \in \mathbb{U}, z_N \in \mathbb{X}_f. \quad (2d)$$

Here $\eta_0 = x(k)$ and we assume that the states and controls are restricted to the domains $\mathbb{X}$ and $\mathbb{U}$, respectively. $\mathbb{X}_f$ is the terminal set and $\mathbb{X}_f \subset \mathbb{X}$. We assume that N is sufficiently long such that $z_N \in \mathbb{X}_f$ is always true for the solution of (2). The set $\mathbb{U}$ is compact and contains the origin; the sets $\mathbb{X}$ and $\mathbb{X}_f$ are closed and contain the origin in their interiors.

The stage cost is given by $\psi(\cdot, \cdot) : \mathfrak{R}^{n_x+n_u} \rightarrow \mathfrak{R}$, while the terminal cost is denoted by $\Psi(\cdot) : \mathfrak{R}^{n_x} \rightarrow \mathfrak{R}$. For tracking problems, we can assume that the states and control variables can be defined with respect to setpoint and reference values, and that the nominal model has the property, $f(0, 0) = 0$.

After solution of (2) the control action is extracted from the optimal trajectory $\{z_0^*, \dots, z_N^*, v_0^*, \dots, v_{N-1}^*\}$ as $u(k) = v_0^*$. At the next time step, the plant evolves as in (1) and we shift the time sequence one step forward, $k = k+1$, obtain the next state estimate $x(k+1)$ and solve the next NMPC problem (2). This recursive strategy gives rise to the feedback law,

$$u(k) = \kappa(x(k)) \quad (3)$$

with $\kappa(\cdot) : \mathfrak{R}^{n_x} \rightarrow \mathfrak{R}^{n_u}$. With the feedback law (3) system (1) becomes

$$\begin{aligned} x(k+1) &= \hat{f}(x(k), \kappa(x(k)), w(k)) \\ &= f(x(k), \kappa(x(k))) + g(x(k), \kappa(x(k)), w(k)) \\ &= f(x(k), \kappa(x(k))) + g(x(k), w(k)) \end{aligned} \quad (4)$$

Hence, we replace $g(x(k), u(k), w(k))$ with $g(x(k), w(k))$ since $u(k) = \kappa(x(k))$. We refer to the above strategy as *ideal NMPC* (iNMPC), where the on-line calculation time is neglected. Correspondingly we denote the feedback law of iNMPC as $u^{id}(k) = \kappa^{id}(x(k))$. iNMPC has well-known stability properties (see [35, 58]) that will be discussed in the next section. On the other hand, the computation time to solve (2) leads to computational delay, which can impact plant performance and

even destabilize the process. To reduce the on-line computational time we consider two crucial elements: a large-scale interior point NLP algorithm for background solution of (2) and an on-line correction step based on a fast, seamless NLP sensitivity step. In the remainder of this section, we consider a reformulated, soft constrained form of problem (2) to remove state variable bounds and inequality constraints, as well as terminal constraints. This reformulation also preserves continuity of the value function with respect to disturbances. We also develop NLP solver and sensitivity tools in the context of primal-dual interior-point solvers.

2.1. NMPC Problem Reformulation

We define $\mathbb{X}$ and $\mathbb{X}_f$ in problem (2) by the inequalities $h(z_l) \leq 0$. These regions are closed and contain the origin in its interior. To develop a robust problem formulation we replace $\mathbb{X}$ and $\mathbb{X}_f$ by ℓ_1 penalty terms and assume, without loss of generality, that $\mathbb{U}$ can be represented by simple upper and lower bounds on v_l .

We also assume that N is sufficiently long such that $z_N \in \mathbb{X}_f$ can always be satisfied. In our case studies, we have used a sufficiently long optimization horizon such that this condition always holds. This horizon length is determined through repeated off-line simulations.¹ As a result, the terminal constraints can be reformulated as terminal cost with penalties in our problem formulation and state constraints appear as penalty terms in the stage costs. Analogously, robust stability of this formulation is also shown in [40] with sufficiently large terminal cost.

To maintain differentiable objective and constraint functions, we reformulate problem (2) to take the following form:

$$\begin{aligned} \min_{z_l, v_l, \xi_l} \quad & (\Psi(z_N) + \rho \xi_N^T e) + \sum_{l=0}^{N-1} (\psi(z_l, v_l) + \rho \xi_l^T e) \\ \text{s.t.} \quad & z_{l+1} = f(z_l, v_l), \quad l = 0, \dots, N-1 \\ & z_0 = x(k) \\ & h(z_l) \leq \xi_l, \xi_l \geq 0; \quad l = 0, \dots, N \\ & v_l \in \mathbb{U}, \quad l = 0, \dots, N-1 \end{aligned} \tag{5}$$

where ξ_l is a slack variable vector and $e = [1, 1, \dots, 1]^T$. If a solution to problem (5) exists, then it is easy to see that the gradients of the equality constraints contain a nonsingular basis matrix, and are therefore linearly independent. Moreover, at

¹Moreover, to check this, we also compared cases with terminal constraints added or removed and found that the results did not change.

the solution $\mathbf{x}^*$ of (5) where $\mathbf{x}^T = [z_0^T \dots, z_N^T, v_0^T, \dots, v_N^T, \xi_1^T \dots, \xi_N^T]$ and with active equality and inequality constraints represented by $c(\mathbf{x}) = 0$, $c_I(\mathbf{x}) \leq 0$, it is straightforward to find a vector $d^T = [d_{z_0}^T, \dots, d_{z_N}^T, d_{v_0}^T, \dots, d_{v_N}^T, d_{\xi_1}^T \dots, d_{\xi_N}^T]$ that satisfies

$$\nabla c(\mathbf{x}^*)^T d = 0, \nabla c_I(\mathbf{x}^*)^T d < 0.$$

Consequently, the Mangasarian-Fromovitz constraint qualification (MFCQ) always holds for problem (5), as shown in [25]. If generalized second order sufficient conditions (GSOSC) are also satisfied, then selecting ρ larger than a finite threshold, $\rho > \bar{\rho}$, will drive ξ_l to zero, where $\bar{\rho}$ is the dual norm of the multipliers at the solution of problem (2). If $\xi_l = 0$, then the solution of (5) is identical to the solution of problem (2). Therefore, stability properties of the soft constrained problem (5) are identical to the hard constrained problem (2). Since a solution with $\xi_l > 0$ for arbitrarily large values of ρ implies that problem (2) is locally infeasible, we assume that a finite $\bar{\rho}$ can be found as long as problem (2) is well-posed. This corresponds to the common assumption that there exists a feasible input sequence, which steers the system to the terminal set. Among other considerations, this requires that the horizon N be long enough to satisfy the terminal conditions.

Moreover, if we apply barrier terms to the inequalities in (5), we obtain an arbitrarily close approximation to the solution of (2) by setting a sufficiently small positive value of μ , as follows:

$$\begin{aligned} \min_{z_l, v_l, \xi_l} \quad & (\Psi(z_N) + \rho \xi_N^T e - \mu \sum_j \ln(\xi_N^{(j)} - h^{(j)}(z_N))) \\ & + \sum_{l=0}^{N-1} (\psi(z_l, v_l) + \rho \xi_l^T e - \mu \sum_j \ln(\xi_l^{(j)} - h^{(j)}(z_l))) \quad (6) \\ \text{s.t.} \quad & z_{l+1} = f(z_l, v_l), \quad z_0 = x(k) \\ & \xi_l \geq 0; v_l \in \mathbb{U}, \quad l = 0, \dots, N-1, \\ & \xi_N \geq 0. \end{aligned}$$

Through the redefinitions: $v_l := [v_l^T, \xi_l^T]^T, l = 1, \dots, N-1$, $\psi(z_l, v_l) := \psi(z_l, v_l) + \rho \xi_l^T e - \mu \sum_j \ln(\xi_l^{(j)} - h^{(j)}(z_l))$ and $\Psi(z_N, \xi_N) := \Psi(z_N) + \rho \xi_N^T e - \mu \sum_j \ln(\xi_N^{(j)} - h^{(j)}(z_N))$

$h^{(j)}(z_N)$), Problem (6) can be rewritten as:

$$\begin{aligned} J_N(x(k)) = \min_{v_l, z_l, \xi_N} \quad & \Psi(z_N, \xi_N) + \sum_{l=0}^{N-1} \psi(z_l, v_l) \\ \text{s. t.} \quad & z_{l+1} = f(z_l, v_l), \quad z_0 = x(k), \quad \xi_N \geq 0, \quad v_l \in \mathbb{U} \quad l = 0, \dots, N-1. \end{aligned} \quad (7)$$

and this replaces problem (2). In the subsequent development we refer to problem (7) as $\mathcal{P}_N(x(k))$. Since the equality constraint gradients are linearly independent and the active bounds for v_l and ξ_N are independent, it is clear that the solution of $\mathcal{P}_N(x(k))$ satisfies the linear independence constraint qualification (LICQ). Along with the presence of bounds as the only inequalities, this property implies the weaker Constant Rank and Mangasarian-Fromovitz Constraint Qualifications (CRCQ and MFCQ), and leads to unique, bounded multipliers. Moreover, if the solutions of (7) also satisfy Second Order Sufficient Conditions (SOSC) then the solution vector, $u(k) = \kappa(x(k))$ and the optimal objective function are continuous in $x(k)$.

In the development and analysis of the robust controllers in Section 3 and beyond, we will assume the slightly stronger assumptions: LICQ, SOSC and *strict complementarity* at the solutions of $\mathcal{P}_N(x(k))$. As detailed in the next section, these conditions guarantee the *differentiability* of the solution vector as well. Based on our experience, these stronger assumptions usually hold as long as the active set is unique at the solution. Indeed, these assumptions are easy to check through successful solution of $\mathcal{P}_N(x(k))$ by the IPOPT solver. In the next section we present the NLP sensitivity background to show how these properties lead to continuity and differentiability of the solution of $\mathcal{P}_N(x(k))$ with respect to disturbances.

2.2. Interior-Point NLP Solvers

To explore continuity and sensitivity properties of problem (7), we represent this problem as:

$$\min_{\mathbf{x}} F(\mathbf{x}, \boldsymbol{\eta}), \text{ s.t. } c(\mathbf{x}, \boldsymbol{\eta}) = 0, \quad x \geq 0 \quad (8)$$

where $\mathbf{x} \in \mathbb{R}^n$ is the variable vector containing the states and controls, and $\boldsymbol{\eta}$ is a *fixed* data vector used to represent uncertain external parameters such as disturbances. The equality constraints are $c(\mathbf{x}, \boldsymbol{\eta}) : \mathbb{R}^n \rightarrow \mathbb{R}^m$. In interior-point solvers, the inequality constraints of problem (8) are handled *implicitly* by adding barrier

terms to the objective function,

$$\min_{\mathbf{x}} \quad F(\mathbf{x}, \eta) - \mu \sum_{j=1}^{n_{\mathbf{x}}} \ln(\mathbf{x}^{(j)}), \quad (9a)$$

$$s.t. \quad \mathbf{c}(\mathbf{x}, \eta) = 0, \quad (9b)$$

where $\mathbf{x}^{(j)}$ denotes the j^{th} component of vector $\mathbf{x}$. Solving (9) for the sequence of $\mu^l \rightarrow 0$, with $l = 0, 1, 2, \dots, \infty$, leads to solution of the original NLP (8). As shown in [17], convergence of solutions of (9) to (8) has been proved under mild conditions (GSOSC and MFCQ).

For a given barrier parameter value μ , IPOPT [52] solves the primal-dual optimality conditions of barrier problems (9) directly,

$$\nabla_{\mathbf{x}} F(\mathbf{x}, \eta) + \nabla_{\mathbf{x}} \mathbf{c}(\mathbf{x}, \eta) \lambda - \mathbf{v} = 0, \quad (10a)$$

$$\mathbf{c}(\mathbf{x}, \eta) = 0, \quad (10b)$$

$$\mathbf{X} \mathbf{V} e = \mu e, \quad (10c)$$

where $\mathbf{X} = \text{diag}(\mathbf{x})$, $\mathbf{V} = \text{diag}(\mathbf{v})$, $e \in \mathfrak{R}^n$ is a vector of ones, and $\lambda \in \mathfrak{R}^m$ and $\mathbf{v} \in \mathfrak{R}^n$ are Lagrange multipliers for the equality constraints and bounds, respectively. The gradient of the objective function is $\nabla_{\mathbf{x}} F(\mathbf{x}, \eta) \in \mathfrak{R}^n$ while $\nabla_{\mathbf{x}} \mathbf{c}(\mathbf{x}, \eta) \in \mathfrak{R}^{n \times m}$ is the constraint Jacobian. To solve this system of nonlinear equations, IPOPT uses an exact Newton method and starts the iteration sequence at point $s_o^T := [\mathbf{x}_o^T \lambda_o^T \mathbf{v}_o^T]$. At the i th Newton iteration, the search direction $\Delta s_i = s_{i+1} - s_i$ is computed by linearization of the KKT conditions (10),

$$\begin{bmatrix} \mathbf{H}_i & \mathbf{A}_i & -I \\ \mathbf{A}_i^T & 0 & 0 \\ \mathbf{V}_i & 0 & \mathbf{X}_i \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_i \\ \Delta \lambda_i \\ \Delta \mathbf{v}_i \end{bmatrix} = - \begin{bmatrix} \nabla_{\mathbf{x}} f(\mathbf{x}_i, \eta) + \mathbf{A}_i \lambda_i - \mathbf{v}_i \\ \mathbf{c}(\mathbf{x}_i, \eta) \\ \mathbf{X}_i \mathbf{V}_i e - \mu e \end{bmatrix} \quad (11)$$

where $\mathbf{A}_i := \nabla_{\mathbf{x}} \mathbf{c}(\mathbf{x}_i, \eta)$. Matrix $\mathbf{H}_i \in \mathfrak{R}^{n \times n}$ is the Hessian of the Lagrange function $\mathcal{L} = F(\mathbf{x}_i, \eta) + \mathbf{c}(\mathbf{x}_i, \eta) \lambda_i - \mathbf{v}_i^T \mathbf{x}_i$. After solving a sequence of barrier problems for $\mu \rightarrow 0$, the solver returns the solution triple $s^{*,T}(\eta) = [\mathbf{x}^{*,T} \lambda^{*,T} \mathbf{v}^{*,T}]$ for problem (8).

Solving the KKT system (11) is the most computationally demanding step in the solution of the NLP. However, the KKT matrix arising from discretized dynamic optimization problems is often very sparse and structured. In IPOPT, after eliminating the bound multipliers from the KKT system (11), we apply a symmetric indefinite factorization of the resulting KKT matrix. The computational

complexity of this strategy is generally favorable (often scaling nearly linearly with problem size), although significant fill-in and computer memory bottlenecks might arise if the sparsity pattern is not properly exploited. In order to factorize the KKT matrix, efficient sparse linear solvers should be applied, e.g. HSL library with the METIS option [12]. Moreover, since the structure of the KKT matrix does not change between iterations, the linear solver needs to analyze the sparsity pattern *only once*.

For sensitivity of the NLP solution, we note that problem (8) is parametric in the data η and the optimal primal and dual variables can be treated as implicit functions of η . For a *sufficiently small* μ^l , the KKT conditions (10) of the barrier problem (9) can be expressed as the equations $\varphi(s^*(\eta), \eta) = 0$ and we denote $\mathbf{K}_*(\eta_0)$ as the KKT matrix in (11). To compute approximate neighboring solutions around an already available nominal solution $s^*(\eta_0)$, we invoke the following classical results.

Theorem 1. (*NLP Sensitivity*) [15] *Assume that $F(\cdot)$ and $c(\cdot)$ of the parametric problem (8) are twice continuously differentiable in a neighborhood of the nominal solution $s^*(\eta_0)$ and this solution satisfies the linear independence constraint qualification (LICQ), strict complementarity and second order sufficient conditions (SOSC), then $s^*(\eta_0)$ is an isolated local minimizer of (8) and the associated Lagrange multipliers are unique. Moreover, for η in a neighborhood of η_0 there exists a unique, continuous and differentiable vector function $s^*(\eta)$ which is a local minimizer satisfying SOSC, strict complementarity and LICQ for (8). Finally, there exists a positive Lipschitz constant L such that $\|s^*(\eta) - s^*(\eta_0)\| \leq L\|\eta - \eta_0\|$ along with a positive Lipschitz constant L_F such that the optimal values $F(\eta)$ and $F(\eta_0)$ satisfy $\|F(\eta) - F(\eta_0)\| \leq L_F\|\eta - \eta_0\|$.*

Since (9) also satisfies LICQ and SOSC and the solution of (9) converges to the solution of (8), Theorem 1 is also true for (9). Therefore, these results allow the application of the implicit function theorem to (10) at $s^*(\eta_0)$ to yield:

$$\mathbf{K}^*(\eta_0) \frac{\partial s^*}{\partial \eta} = - \frac{\partial \varphi(s(\eta, \cdot), \eta)}{\partial \eta} \Big|_{s^*(\eta_0), \eta_0} \quad (12)$$

where $\mathbf{K}^*(\eta_0)$ is the KKT matrix of (10) evaluated at $s^*(\eta_0)$. If the nominal solution satisfies SOSC and LICQ, then the KKT matrix is non-singular [38] and can be used to compute the sensitivity matrix from (12). With this, solutions and

their first-order estimates of neighboring problems can be obtained from:

$$s^*(\eta) = s^*(\eta_0) + \frac{\partial s^*}{\partial \eta}(\eta - \eta_0) + O(|\eta - \eta_0|^2) \quad (13)$$

$$\tilde{s}(\eta) = s^*(\eta_0) + \frac{\partial s^*}{\partial \eta}(\eta - \eta_0) \quad (14)$$

where $\tilde{s}(\eta)$ is an approximate solution of $s^*(\eta)$. From continuity and differentiability of the optimal solution vector, there exists a positive Lipschitz constant L_q such that,

$$|\tilde{s}(\eta) - s^*(\eta)| \leq L_q |\eta - \eta_0|^2. \quad (15)$$

The step $\Delta s(\eta) = \tilde{s}(\eta) - s^*(\eta_0)$ in (14) can also be found by a first order difference approximation of the KKT conditions (10) at $s^*(\eta_0)$ to yield:

$$\varphi(s^*(\eta_0), \eta_0) + (\varphi(s^*(\eta_0), \eta) - \varphi(s^*(\eta_0), \eta_0)) + \mathbf{K}^*(\eta_0) \Delta s(\eta) = \varphi(s^*(\eta), \eta).$$

$\varphi(s^*(\eta), \eta) = 0$ and $\varphi(s^*(\eta_0), \eta_0) = 0$, since they are at KKT points, and we obtain:

$$\mathbf{K}^*(\eta_0) \Delta s(\eta) = -\varphi(s^*(\eta_0), \eta). \quad (16)$$

where the right-hand side corresponds to the KKT conditions (10) evaluated at the nominal solution with η_0 as the parameter value. The second argument of φ is the parameter whose value changes after the optimization problem is solved. The optimization problem is solved with η_0 as the value of the parameter, and the new parameter value is η . Here, $\Delta s(\eta) = \tilde{s}(\eta) - s^*(\eta_0)$ is a Newton step taken from $s^*(\eta_0)$ towards the solution of a neighboring problem (8) so that $\tilde{s}(\eta)$ satisfies (14)-(15). Also, for the step length $\tau \in [0, 1]$, the quantity $\hat{s} = s^*(\eta_0) + \tau \Delta s(\eta)$ has the property:

$$|\hat{s} - s^*(\eta)| \leq (1 - \tau) \left| \frac{\partial s^*}{\partial \eta} \right| |\eta - \eta_0| + L_q |\eta - \eta_0|^2. \quad (17)$$

This result will be used in the next section to analyze the clipping strategy.

2.3. Active Set Changes

When the perturbation $\eta - \eta_0$ is large enough to induce a change in the active constraint set at the solution of (8), tracking the KKT conditions becomes nonsmooth, the linearization (16) is no longer valid and the approximate solution $\tilde{s}(\eta)$ may not even be feasible to $\mathbb{X}$ and $\mathbb{U}$. Since Theorem 1 no longer holds at the

point where the transition in active sets occurs, $s^*(\eta)$ is no longer differentiable (and may not even be continuous) with respect to η . However, under Generalized Second Order Sufficient Conditions (GSOSC) and a relaxed set of constraint qualifications (CRCQ and MFCQ) (satisfied by $\mathcal{P}_N(x)$), one can still obtain continuity of $s^*(\eta)$ and the value function with respect to η as well as the *directional derivative* of $\mathbf{x}^*(\eta)$ for $\eta - \eta_0$.

As developed in [44], this directional derivative is determined by solving a specialized QP that can be viewed as an extension of (16), with linearized inequality constraints included and strongly active inequalities (with positive multipliers) expressed as equality constraints. Because multipliers at the active set transition are nonunique, a linear program (LP) in dual space must first be solved to maximize the linearized prediction of the Lagrange function. In [25] we apply this formulation to develop a path-following algorithm to track $\mathbf{x}^*(\eta)$ with respect to η . Detailed presentation of this path-following approach and general underlying concepts that govern NLP sensitivity are also included in [25]. While this approach leads to a rigorous treatment of NLP sensitivity, it is more expensive than the simple update (16).

Instead of path-following, we can apply a cheaper strategy called “clipping in the first interval,” where we perturb the solution only up to the active set change so that Theorem 1 still holds. Here, feasibility of the soft constrained formulation (7) only requires $v_l \in \mathbb{U}$, and clipping ensures that the perturbed control variable value, $u(k)$ remains within its bounds. Because the clipping strategy requires no additional computational cost beyond (16), we incorporate this approach within the advanced step NMPC developed and analyzed in the following section, and applied in the distillation case study in Section 4. Moreover, in [25, 56] a comparison of the simple clipping approach with more detailed approaches shows very good performance in the presence of active set changes. Additional discussion of these strategies can be found in [25, 56].

3. Advanced-Step NMPC Strategies

Treatment of NMPC problem $\mathcal{P}_N(\eta_0)$ with the above NLP and sensitivity tools corresponds to the off-line and on-line components, respectively, of advanced step strategies. At time t_k we use the current estimate $x(k)$ and control $u(k)$ to predict the future state at t_{k+1} , but we also begin the execution of problem (7) at t_k . At the solution of (7) we obtain the KKT matrix $\mathbf{K}^*$.

Once the state estimate $x(k+1)$ becomes available, we compute a fast back-solve with $\mathbf{K}^*$ to obtain the control action $u(k+1)$. Consequently, the proposed

framework for the advanced-step NMPC (asNMPC) strategy, with clipping to ensure $u(k+1) \in \mathbb{U}$, can be summarized as follows.

In background, between t_k and t_{k+1} :

1. Use $x(k)$ and $u(k)$ to predict the future state through $\bar{x}(k+1) \equiv f(x(k), u(k))$. Set $\eta_0 = \bar{x}(k+1)$ and solve the *predicted* problem $\mathcal{P}_N(\bar{x}(k+1))$.
2. At the solution $s^*(\eta_0)$, retain factors of $\mathbf{K}^*(\eta_0)$.

On-line, at t_{k+1} :

1. Obtain the state estimate $x(k+1)$ and set $\eta = x(k+1)$. Compute the sensitivity step $\Delta s(\eta)$ from (16). Extract the appropriate element of $\Delta s(\eta)$, called Δv_0 which corresponds to the perturbed control variable at $l = 0$.
 - Calculate $\tilde{v}_0 = v_0^* + \Delta v_0$. Note that elements of v_0^* that are strongly active have zero perturbations.
 - If $\tilde{v}_0 \in \mathbb{U}$, set $\tau = 1$.
 - Else, find the largest value of $\tau \in [0, 1]$ such that $v_0^* + \tau \Delta v_0 \in \mathbb{U}$.
2. Set $u(k+1) = v_0^* + \tau \Delta v_0 \in \mathbb{U}$, and return to the background phase with $k := k+1$.

Stability and robustness properties of the advanced step strategy have been analyzed in [57] for the case without clipping (i.e., $\tau = 1$). In the remainder of this section we restate and modify these results to show that the added clipping strategy also leads to input to state stability.

3.1. Nominal and ISS Stability of asNMPC

We begin with well-known results on nominal and robust stability for the iN-MPC controller $u^{id} = \kappa^{id}(x)$ [35, 29].

Assumption 2. (*Nominal Stability Assumptions of iNMPC*)

- The terminal penalty $\Psi(\cdot)$, satisfies $\Psi(z) > 0, \forall z \in \mathbb{X}_f \setminus \{0\}$,
- There exists a local control law $u = h_f(z)$ defined on $\mathbb{X}_f$, such that $f(z, h_f(z)) \in \mathbb{X}_f, \forall z \in \mathbb{X}_f$, and $\Psi(f(z, h_f(z))) - \Psi(z) \leq -\psi(z, h_f(z)), \forall z \in \mathbb{X}_f$.

- The optimal stage cost $\psi(x, u) = \psi(x, \kappa(x))$ satisfies $\alpha_p(|x|) \leq \psi(x, u) \leq \alpha_q(|x|)$ where $\alpha_p(\cdot)$ and $\alpha_q(\cdot)$ are $\mathcal{K}$ functions.

Nominal stability of iNMPC can be paraphrased by the following theorem.

Theorem 3. (Nominal Stability of iNMPC [46]) Consider the moving horizon problem (7) and associated control law $u = u^{id}$, that satisfies Assumption 2. Then, $J_N(x)$ from $\mathcal{P}_N(x)$ is a Lyapunov function and the closed-loop system is asymptotically stable.

In the nominal case both asNMPC and iNMPC controllers produce identical control actions $u^{id}(k+1) = u^{as}(k+1)$ (because $g(x, w) = 0$ and $\Delta s(\eta) = 0$). Nominal stability of asNMPC then follows directly from Theorem 3 [57].

For the analysis of robust stability properties of the iNMPC we consider Input-to-State Stability (ISS) [35, 27].

Definition 4. (Input-to-State Stability)

- The system (1) is ISS in $\mathbb{X}$ if there exists a $\mathcal{KL}$ function β , and a $\mathcal{K}$ function γ such that for all w in the bounded set $\mathcal{W}$,

$$|x(k)| \leq \beta(|x(0)|, k) + \gamma(|w|), \forall k \geq 0, \forall x(0) \in \mathbb{X} \quad (18)$$

- A function $V(\cdot)$ is called an ISS-Lyapunov function for system (1) if there exist a set $\mathbb{X}$, $\mathcal{K}$ functions $\alpha_1, \alpha_2, \alpha_3$ and σ such that, $\forall x \in \mathbb{X}$ and $\forall w \in \mathcal{W}$, we have:

$$\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|) \quad (19a)$$

$$V(\hat{f}(x, u, w)) - V(x) \leq -\alpha_3(|x|) + \sigma(|w|) \quad (19b)$$

Moreover, if $\mathbb{X}$ is a robustly invariant set for system (1) and $V(\cdot)$ is an ISS-Lyapunov function for this system, then the resulting system is ISS in $\mathbb{X}$ [8, 35].

There are different forms of $g(x, w)$. For example, $g(x, w)$ could be additive disturbances, or plant-model mismatch. An approach to handle more general perturbations including state dependent ones was proposed and analyzed in [41]. Also, as shown in [19], ISS may be lost if there are state constraints, or terminal constraints coupled with short optimization horizons. This necessarily leads to a discontinuous feedback law and cost function. Also, the results in [30] reveal a relation between existence of a continuous Lyapunov function and inherent input-to-state stability.

As developed in Section 2.1, we address this problem by replacing state constraints with penalties in the stage costs. It is pointed out in [40] that when there are no state constraints, the terminal constraint could be replaced by an appropriate terminal penalty without loss of robustness. Also, we need to specify a sufficiently large prediction horizon (determined through off-line simulation) such that $x \in \mathbb{X}_f$ is satisfied and is not active. This satisfies the assumptions for Theorems 3 and 6, so the results still hold here. Moreover, if we assume that SOSC, LICQ and strict complementarity hold at the optimal solution of $\mathcal{P}_N(x)$ (problems (7), (8)), the value function and feedback law are continuous and differentiable. As a result, robustness will not be lost.

To analyze the robustness of the asNMPC controller, we recognize that with $\bar{x}(k+1) = f(x(k), u(k))$, there exists a future mismatch $x(k+1) - \bar{x}(k+1) = g(x(k), w(k))$ at the next time step, giving rise to two different problems $\mathcal{P}_N(\bar{x}(k+1))$ and $\mathcal{P}_N(x(k+1))$, with optimal costs $J_N(\bar{x}(k+1))$ and $J_N(x(k+1))$, respectively. To account for this, we define the mismatch term,

$$\varepsilon(x(k+1)) := J_N(x(k+1)) - J_N(\bar{x}(k+1)). \quad (20)$$

Under Theorem 1, there exists a local positive Lipschitz constant L_J such that $\forall x \in \mathbb{X}$,

$$\varepsilon(x(k+1)) \leq L_J |g(x(k), w(k))|. \quad (21)$$

We therefore apply the following assumptions for the ideal and advanced step NMPC controllers.

Assumption 5. (*Robust Stability Assumptions*)

- The solution of $\mathcal{P}_N$, given by $s^*(\eta)$ in problem (8) satisfies LICQ, SOSC and strict complementarity. From Theorem 1, the objective function and $s^*(\eta)$ are therefore continuous and differentiable with respect to η and the resulting feedback law, derived from $s^*(\eta)$, can be represented as $u = \kappa(x)$.
- $g(x, w)$ is Lipschitz with respect to all its arguments with Lipschitz constant L_g : $|g(x, w)| \leq |g(x, 0)| + L_g |w|$, where

$$|g(x, 0)| \leq \alpha_0(|x|), \quad (22)$$

and $\alpha_0(|x|)$ is a $\mathcal{K}$ function.

Robust stability of the iNMPC controller can be established from the following theorem.

Theorem 6. (*Robust ISS Stability of iNMPC (Theorem 2 in [35], see also [27, 57]) Under Assumptions 2 and 5 with $\alpha_0(|x|) \leq \frac{\rho}{L_J} \alpha_p(|x|)$ and $\rho \in (0, 1)$ is an arbitrary real number, the cost function $J_N(x)$ obtained from the solution of $\mathcal{P}_N(x)$ (7) is an ISS-Lyapunov function and the resulting closed-loop system is ISS stable.*)

For the analysis of the robustness properties of the asNMPC controller we need to consider the effect of NLP sensitivity errors. The forward simulation $\bar{x}(k+1) = f(x(k), u(k))$ predicts the future state at t_{k+1} but the plant will evolve with uncertain dynamics generating $x(k+1)$. Moreover, we need to distinguish between iNMPC using $u^{id}(k)$ and asNMPC, which generates $u^{as}(k) = \kappa^{as}(x(k))$. To interpret this difference we consider an extended problem $\mathcal{P}_{N+1}(x(k), \hat{u}(k))$:

$$\hat{J}(x(k), \hat{u}(k)) := \min_{z_l, v_l} \quad \Psi(z_N) + \psi(x(k), \hat{u}(k)) + \sum_{l=0}^{N-1} \psi(z_l, v_l) \quad (23a)$$

$$\text{s.t. } z_{l+1} = f(z_l, v_l) \quad l = 1, \dots, N-1 \quad (23b)$$

$$z_0 = f(x(k), \hat{u}(k)) \quad (23c)$$

$$v_l \in \mathbb{U} \quad (23d)$$

This problem has an equivalent solution to problem $\mathcal{P}_N(z_0)$ and we consider $\hat{J}(x, \hat{u})$ as our candidate ISS Lyapunov function. We define $J^{as}(x(k)) := \hat{J}(x(k), u^{as}(k))$, $J^{id}(x(k)) := \hat{J}(x(k), u^{id}(k))$, and also $J^{id}(\bar{x}(k)) := \hat{J}(\bar{x}(k), \bar{u}^{id}(k))$, where $u^{id}(k)$ and $\bar{u}^{id}(k)$ are determined as variables, $\hat{u}(k) \in \mathbb{U}$, in $\mathcal{P}_{N+1}$. For the next time step, we define the following residuals as:

$$\varepsilon_s(x(k+1)) := J^{id}(x(k+1)) - J^{id}(\bar{x}(k+1)) \quad (24a)$$

$$\varepsilon_{as}(x(k+1)) := J^{as}(x(k+1)) - J^{id}(x(k+1)) \quad (24b)$$

where ε_s accounts for the model mismatch as in (20) while ε_{as} accounts for approximation errors introduced by NLP sensitivity. From (17) and Theorem 1 we have positive Lipschitz constants L_J, L_u, L_{sv} and L_{sq} such that $\forall x \in \mathbb{X}$,

$$\varepsilon_s(x(k+1)) \leq L_J(|x(k+1) - \bar{x}(k+1)| \leq L_J|g(x(k), w(k))|) \quad (25a)$$

$$\begin{aligned} \varepsilon_{as}(x(k+1)) &\leq L_u(|u^{as}(k+1) - u^{id}(k+1)|) \\ &\leq L_J((1-\tau)L_{sv} + L_{sq}|g(x(k), w(k))|) \cdot |g(x(k), w(k))| \end{aligned} \quad (25b)$$

By comparing the successive costs $J^{as}(x(k))$ and $J^{as}(x(k+1))$, we arrive at a similar ISS property as in Theorem 6. However, for this result we require an additional assumption that limits the choice of mismatch function $|g(x, 0)|$, which must be bounded above by g_{max} . For this additional assumption on $g(x, 0)$, note that a bound on $|x|$ is not implied, because $|g(x, 0)| \leq \alpha_0(|x|) \leq g_{max}$ is *not* assumed.

Theorem 7 (Robust Stability of asNMPC). *Given Assumptions 2 and 5 with*

$$\alpha_0(|x|) \leq \frac{\rho}{\zeta} \alpha_p(|x|) \text{ where } \zeta = L_J(1 + L_{sv} + L_{sq}g_{max}),$$

$\rho \in (0, 1)$ is an arbitrary real number, and $|g(x, 0)|$ is bounded above by g_{max} . Under these conditions, the cost function $J^{as}(x)$, obtained from the solution of Problem (23) with $u = u^{as}$, is an ISS-Lyapunov function and the resulting closed-loop system is ISS stable.

Proof : We compare the costs $J^{as}(x(k))$, $J^{as}(x(k+1))$ and use the mismatch terms in (25a)-(25b) to obtain,

$$\begin{aligned} & J^{as}(x(k+1)) - J^{as}(x(k)) \\ &= J^{id}(\bar{x}(k+1)) - J^{as}(x(k)) + J^{id}(x(k+1)) - J^{id}(\bar{x}(k+1)) \\ & \quad + J^{as}(x(k+1)) - J^{id}(x(k+1)) \\ &\leq -\psi(x(k), u^{as}(k)) + \varepsilon_s(x(k+1)) + \varepsilon_{as}(x(k+1)) \\ &\leq -\alpha_p(|x(k)|) + \varepsilon_s(x(k+1)) + \varepsilon_{as}(x(k+1)). \end{aligned} \tag{26}$$

The last two inequalities follow by noting that the solution of Problem (23) at k provides a feasible solution to Problem (23) at $k+1$, and from Assumption 2. Substituting the bounds (25a)-(25b) for the error terms leads to:

$$\begin{aligned} & \varepsilon_s(x(k+1)) + \varepsilon_{as}(x(k+1)) \\ &\leq L_J(1 + (1 - \tau)L_{sv} + L_{sq}(|g(x(k), 0)| + L_g|w(k)|))(|g(x(k), 0)| + L_g|w(k)|)) \\ &\leq L_J[(1 + L_{sv} + L_{sq}g_{max})\alpha_0(|x|) \\ & \quad + L_g(1 + L_{sv} + (1 + L_{sq})g_{max})|w(k)| + (L_g|w(k)|)^2] \\ &\leq L_J(1 + L_{sv} + L_{sq}g_{max})(\rho/\zeta)\alpha_p(|x|) \\ & \quad + L_J(L_g(1 + L_{sv} + (1 + L_{sq})g_{max})|w(k)| + (L_g|w(k)|)^2) \\ &\leq \rho\alpha_p(|x(k)|) + \sigma|w(k)| \end{aligned}$$

where the first two inequalities follow from Assumption 5 and from $\tau \geq 0$, and the last inequality follows from $\zeta = L_J(1 + L_{sv} + L_{sq}g_{max})$ and $\sigma(|w|) = L_J(L_g(1 + L_{sv} + (1 + L_{sq})g_{max})|w(k)| + (L_g)^2|w(k)|^2)$. The theorem is proved by substituting this result into (26) to yield:

$$J^{as}(x(k+1)) - J^{as}(x(k)) \leq (\rho - 1)\alpha_p(|x(k)|) + \sigma|w(k)|. \quad \square$$

Note that ζ determines the bound on $g(x, 0)$ in the ISS condition (22) and we can identify the following performance levels.

- If clipping ($\tau < 1$) is applied at frequent intervals as $k \rightarrow \infty$, we have the largest ζ and the smallest values of $|g(x, 0)|$ allowed for robustness due to (22).
- If $\tau = 1$ for all $k \geq k_1 > 0$ then we have the robust performance of unconstrained asNMPC derived in [57], whose additional loss is due to a term proportional to $|g(x(k), w(k))|^2$.
- If $g(x(k), w(k)) = 0$, then $x(k+1) = \bar{x}(k+1)$ and asNMPC recovers the nominal stability and performance of iNMPC, and clipping is not needed.

4. The amsNMPC Formulation

The asNMPC strategy requires problem (7) to be solved within one sampling time. If solution of problem (7) requires $N_s \geq 1$ sampling times, one could clearly update the control variables less frequently, and thus detune the controller, but performance of the controller would certainly suffer. For this case, *advanced multi-step NMPC* (amsNMPC) methods are proposed in [56]; these methods take two forms. The *serial approach* performs a sensitivity update of the control variable at every sampling time but solves the NLP problem (7) over $N_s \geq 1$ sampling times. On the other hand, the *parallel approach* is applied over N_s processors and initiates the solution of problem (7) at every sampling time. When the controller receives an updated state estimate, the solution of the most recent NLP solution receives a sensitivity correction (16) to obtain the corresponding control variable, and the free processor is applied to initiate a new NLP problem. The cycle then repeats as NLP solutions and free processors are obtained. The algorithms, equipped with clipping in the first control interval, proceed as follows.

Parallel Approach

When N_s sampling times are required to solve the NLP problem, we project problem (7) N_s sampling times in advance to determine the predicted control variables. Also, $v_{0|k}$ is updated from (16) once the new state estimate $x(k)$ is obtained and we define $z_{N_s|k}$ as the N_s th predicted state from the NLP solution, given the current state $x(k)$. The updated $z_{N_s|k}$ is also the prediction that becomes the initial value for the next NLP problem. Once the control variable is updated through (16) at t_{k+1} , the next available parallel processor is applied to deal with the next background NLP $\mathcal{P}_N(z_{N_s|k+i})$. The parallel approach is implemented as follows:

For $i = 0 : N_s - 1$,

On-line: at t_{k+i} , having $x(k+i)$, update $v_{0|k+i}$ and $z_{N_s|k+i}$ using $s^* + \tau\Delta s$ from (16). Inject the updated $v_{0|k+i} \in \mathbb{U}$ as $u(k+i)$ to the plant.

Background: take the updated $z_{N_s|k+i}$ as the initial value and solve the background NLP problem ($\mathcal{P}_N(z_{N_s|k+i})$) using a new processor. Set $k = k + N_s$ and repeat the cycle.

More details of this approach can be found in [56].

Serial Approach

Again, N_s sampling times are needed to solve the NLP problem. Suppose the optimal solution of the last NLP solution is known at t_k . Knowing $x(k)$ and $u(k)$ we predict $z_{N_s|k}$ as an initial value and solve problem $\mathcal{P}_N(z_{N_s|k})$ between t_k and t_{k+N_s} . In the meantime, the current control variables $v_{1|k}, v_{2|k}, \dots, v_{N_s-1|k}$ are updated on-line using the sensitivity ($s^* + \tau\Delta s$), based on solution of the previous NLP problem. However, for these sensitivity-based updates, (16) cannot be applied directly. Instead an augmented sensitivity system is constructed to account for the predicted states and controls between t_k and t_{k+N_s} . This augmented KKT sensitivity system is solved via Schur complement decomposition and is described in [56].

The serial amsNMPC strategy is executed as follows:

Background: At t_k , having $x(k)$ and $u(k)$, update $v_{0|k}$ and $z_{N_s|k}$ using $s^* + \tau\Delta s$ from (16). Solve problem $\mathcal{P}_N(z_{N_s|k})$ between t_k and t_{k+N_s} with an updated $z_{N_s|k}$ as the initial value.

On-line: for $i = 1 : N_s - 1$,

At t_{k+i} , having $x(k+i)$, update the augmented sensitivity system and obtain $v_{i|k} \in \mathbb{U}$ using $s^* + \tau \Delta s$. Inject the updated $v_{i|k} \in \mathbb{U}$ as $u(k+i)$ to the plant. Set $k = k + N_s$ and repeat the cycle.

Both approaches are closely related as they solve problem (7) N_s steps in advance using the predicted states at $k + N_s$ from the current optimal solution. Moreover, as shown in [56], both satisfy nominal stability properties under mild assumptions. On the other hand, the serial approach uses only one processor, where (7) is solved every N_s sampling times, and the first N_s control variables in the horizon are updated through an augmented sensitivity approach. The parallel approach uses multiple processors and problem (7) is initiated at every sampling time, with the first control variable updated using (16), but with different KKT matrices at every sampling time. With either approach, amsNMPC generally reduces on-line computational cost by two to three orders of magnitude, as with the asNMPC approach. We also note that amsNMPC with $N_s = 0$, $N_s = 1$ and $N_s = \infty$ corresponds to ideal NMPC, asNMPC and the basic neighboring extremal update (NEU) approach [54], respectively.

4.1. C3 Splitter Column Case Study

To demonstrate the amsNMPC approach, we consider a large-scale propane-propylene distillation column, also known as a C3 splitter. Modeled on an actual refinery unit [16], the binary distillation column consists of $N_T = 158$ trays with feed on the 43rd tray, a total condenser, and a thermosyphon reboiler. The distillation model consists of dynamic mass balance, equilibrium, summation and heat balance (MESH equations) for each tray along with vapor-liquid equilibrium, and tray hydraulics as follows:

$$\begin{aligned}
\frac{dM_i}{dt} &= F_i + L_{i-1} - L_i + V_{i+1} - V_i \\
\frac{d(M_i x_i)}{dt} &= F_i x_F + L_{i-1} x_{i-1} - L_i x_i + V_{i+1} y_{i+1} - V_i y_i \\
y_{i,j} &= K_{i,j}(T_i, P_i, x_i) x_{i,j} \\
0 &= \sum_{j \in \text{COMP}} y_{i,j} - \sum_{j \in \text{COMP}} x_{i,j} \\
\frac{d(M_i h_i)}{dt} &= F_i h_F + L_{i-1} h_{i-1}^L - L_i h_i^L + V_{i+1} h_{i+1}^V - V_i h_i^V \\
i &= 1, \dots, N_T
\end{aligned} \tag{27}$$

where $COMP$ is the set of components; M_i, F_i, x_i, y_i are the liquid holdup, feed flowrate, liquid composition and vapor composition on the i th tray respectively; x_F is the feed composition; L_i, V_i are the liquid and vapor leaving the i th tray respectively; T_i, P_i are the temperature and pressure on each tray respectively; h_i^L, h_i^V, h_F are the liquid, vapor and feed enthalpy respectively. For the propane-propylene system, equilibrium constants are determined from DePriester nomographs. A detailed description of these equations, along with its reformulation from index 2 to index 1, can be found in [16]. Negligible vapor holdup and constant pressure drop from tray to tray are assumed.

The controlled states are the concentrations of propylene of the first and last trays, while the control variables are the steam pressure in the reboiler and the bottoms flow rate. The objective is to keep the controlled states at their setpoints, so the objective function is composed of quadratic tracking terms. As a result, the first part of Assumption 2 is satisfied. We tried different horizon lengths to assure that the solution $u = h_f(z)$ exists and converges to the optimum, and to assure that the second part of Assumption 2 is satisfied as well. Since the solution exists, the third part of Assumption 2 is satisfied. The horizon length we use is $N = 35$. This case study compares the performance of ideal NMPC and the parallel approach of amsNMPC. After discretizing the ordinary differential equations with orthogonal collocation, the NLP problem has 111650 variables and 111580 constraints. Solution of problem (7) requires 90 CPU seconds, but with a sampling time of only 60 s, N_s must be greater than one, and amsNMPC must be used. On the other hand, the on-line computation (solving (16)) requires less than 1 CPU second.

To show the difference in performance with different N_s , we assume we can choose $N_s = 0$ (iNMPC), $N_s = 1$ (asNMPC) as well as $N_s = 2$ and $N_s = 3$. For this case, we change the setpoint at $t = 30$ and introduce Gaussian noise with a standard deviation of 1% of the setpoint on all outputs but $x[Ntray]$. If this level of noise is added to $x[Ntray]$, its upper bound will be violated, so only 0.1% noise is added to $x[Ntray]$. The performance of ideal NMPC and the parallel approach with $N_s = 1, 2, 3$ are compared. The noise level is then increased to 3% and 5%. For each level, the noise is generated randomly only once and then the same noise sequence is used by iNMPC, asNMPC and the parallel approach, so that all results are consistent. Fig.1 shows that with 1% of noise, set points are tracked very well, and there is not much difference among different cases, in the state profile as well as the control profile. Increasing the noise level to 3% in Fig.2, shows that the difference in state and control profiles becomes larger as noise level increases. With 5% additive noise, Fig.3 shows that the difference is even more significant. For large noise levels, we would expect the performance

(a) State profiles

(b) Control profiles

Figure 1: 1% additive noise. The results of ideal NMPC are shown in gray; the results of the parallel amsNMPC are shown in blue, red and green with $N_s = 1, 2, 3$.

(a) State profiles

(b) Control profiles

Figure 2: 3% additive noise. The results of idNMPC are shown in gray; the results of the parallel amsNMPC are shown in blue, red and green with $N_s = 1, 2, 3$.

(a) State profiles

(b) Control profiles

Figure 3: 5% additive noise. The results of ideal NMPC are shown in gray; the results of the parallel amsNMPC are shown in blue, red and green with $N_s = 1, 2, 3$.

of amsNMPC to deteriorate as N_s increases, because the linearization used for the sensitivity update becomes less accurate. On the other hand, if the model responses to noise are not highly nonlinear, then performance loss with increasing N_s does not occur. We observe this effect in this case study; the state profiles generated with different N_s remain similar in all noise cases.

5. Economic NMPC and Dynamic Real-Time Optimization

As explored in [6], combining the two-layer RTO approach into a single dynamic optimization may lead to significant improvements in process performance. In particular, *artificial setpoints* used in (7) and determined from a steady state optimization are no longer needed. Instead, a dynamic optimization directly maximizes an economic objective using a well-tuned dynamic process model. This single stage D-RTO thus has advantages over conventional RTO because the NMPC profile is driven by economic stage costs instead of mere tracking terms. As a result, this usually leads to better economic performance. In fact, significant differences between tracking to optimal steady set points and economic NMPC were observed in the case studies in [2], even for nominal cases.

However, economic NMPC or, equivalently, dynamic real-time optimization, is not as simple as replacing the stage costs in (7); stability and robustness properties are still required. Moreover, since the setpoint, i.e., the “zero” in Theorem 1, is unknown, we need to consider how stability can be guaranteed.

The most direct way to enforce stability of economic NMPC is to consider an infinite time horizon, $N \rightarrow \infty$. Significant early advances have been made in designing NMPC controllers along these lines [3, 55], although many issues relating to computation and robust stability remain. A more desirable result would be to assume that Economic NMPC drives the process to a steady state. However, this assumption cannot be guaranteed and counterexamples have been presented in [3].

Nevertheless, there are several reasons to enforce convergence to a steady state. In particular, production planning with economic models over long time scales is based on steady state models, and consistency with these models must be ensured. Also, plant stability and robustness are easier to analyze under steady state assumptions. Finally, steady state (or cyclic steady state) operation is easier to monitor, analyze and manage.

To ensure that Economic NMPC converges to steady state we revisit NMPC stability analysis through the following constructions from [10]. We first define the steady state problem and establish a suitable Lyapunov function by subtracting

the optimal steady state from the original system and adding a *rotated* term. Here, the original system is asymptotically stable at the optimum if the rotated system is asymptotically stable at the origin. In addition, we observe that the rotated Lyapunov function is strictly decreasing.

To define implicit reference values for the states and controls, we consider the steady state optimization problem given by:

$$\min \psi(z, v), \text{ s.t. } z = f(z, v), z \in \mathbb{X}, v \in \mathbb{U} \quad (28)$$

with the solution given by (z^*, v^*) . We introduce a transformed system by subtracting the optimal steady state from the predicted values as follows:

$$\bar{z}_l = z_l - z^*, \quad \bar{v}_l = v_l - v^* \quad (29)$$

and the transformed state evolves according to

$$\bar{z}_{l+1} = \bar{f}(\bar{z}_l, \bar{v}_l) = f(\bar{z}_l + z^*, \bar{v}_l + v^*) - z^* \quad (30)$$

and $\bar{z}_l \in \bar{\mathbb{X}}$ and $\bar{v}_l \in \bar{\mathbb{U}}$, where $\bar{\mathbb{X}}$ and $\bar{\mathbb{U}}$ are the corresponding sets for the transformed system. From equation (30), we see that when $(\bar{z}_l, \bar{v}_l) = (0, 0)$,

$$z_{l+1}^* = f(z_l^*, v_l^*) = z^* \quad (31)$$

Similarly, we define the transformed stage and terminal costs as:

$$\bar{\psi}(\bar{z}_l, \bar{v}_l) = \psi(\bar{z}_l + z^*, \bar{v}_l + v^*) - \psi(z^*, v^*) \quad (32)$$

$$\bar{\Psi}(\bar{z}_N) = \Psi(\bar{z}_N + z^*) - \Psi(z^*) \quad (33)$$

and $\bar{\Psi}(0) = \bar{\psi}(0, 0) = 0$, so that the iNMPC subproblem is now given by:

$$\begin{aligned} \min \quad & \bar{\Psi}(\bar{z}_N) + \sum_{l=0}^{N-1} \bar{\psi}(\bar{z}_l, \bar{v}_l) \\ \text{s.t.} \quad & \bar{z}_{l+1} = \bar{f}(\bar{z}_l, \bar{v}_l), \quad l = 0, \dots, N \\ & \bar{z}_0 = x(k) - x^*(k), \bar{z}_l \in \bar{\mathbb{X}}, \bar{v}_l \in \bar{\mathbb{U}}, \bar{z}_N \in \bar{\mathbb{X}}_f, \end{aligned} \quad (34)$$

As in Section 2.1, we reformulate problem (34) to the form of (7) by removing the terminal constraints and replacing state constraints by penalty terms.

A key concern is that Assumption 2 generally does not hold for transformed economic stage costs $\bar{\psi}$. Instead, the *rotated* stage and terminal costs from [10, 25]

are given by:

$$L(\bar{z}_l, \bar{v}_l) = \bar{\Psi}(\bar{z}_l, \bar{v}_l) + \lambda^{*,T}(\bar{z}_l - \bar{f}(\bar{z}_l, \bar{v}_l)) \quad (35)$$

$$L_f(\bar{z}) = \bar{\Psi}(\bar{z}) + \lambda^{*,T}\bar{z}. \quad (36)$$

where λ^* is the multiplier from the equality constraints in (28). As shown in [10, 25] substitution of the rotated costs into (34) does not alter the optimal solution of (34). Moreover, the rotated costs have a local minimum at $(0, 0)$. If $L(\bar{z}, \bar{v})$ and $L_f(\bar{z})$ are also strongly convex functions, they have a global minimum at $(0, 0)$ and satisfy Assumption 2. To see this, we note $\bar{f}_i(0, 0) = 0$, $L(\bar{z}, \bar{v}) \geq 0$, $L(0, 0) = 0$, $[\bar{z}^T \bar{v}^T] \nabla L(0, 0) \geq 0$ and we can write:

$$\begin{aligned} L(\bar{z}, \bar{v}) &= \nabla L(0, 0)^T \begin{bmatrix} \bar{z} \\ \bar{v} \end{bmatrix} + 1/2 \int_0^1 [\bar{z}^T \bar{v}^T] \nabla^2 L(\tau \bar{z}, \tau \bar{v}) \begin{bmatrix} \bar{z} \\ \bar{v} \end{bmatrix} d\tau \\ &\geq 1/2 \int_0^1 [\bar{z}^T \bar{v}^T] \nabla^2 L(\tau \bar{z}, \tau \bar{v}) \begin{bmatrix} \bar{z} \\ \bar{v} \end{bmatrix} d\tau. \end{aligned} \quad (37)$$

If $L(\bar{z}_i, \bar{v}_i)$ is strongly convex, we have for some $\gamma_L > 0$,

$$\begin{aligned} L(\bar{z}, \bar{v}) &\geq 1/2 \int_0^1 [\bar{z}^T \bar{v}^T] \nabla^2 L(\tau \bar{z}, \tau \bar{v}) \begin{bmatrix} \bar{z} \\ \bar{v} \end{bmatrix} d\tau \\ &\geq \gamma_L(|\bar{z}|^2 + |\bar{v}|^2) \geq \alpha_p(|\bar{z}|) \end{aligned} \quad (38)$$

and Assumption 2 holds. Note that strong convexity of L involves properties of both the cost and model functions.

On the other hand, if (38) does not hold, regularization terms must be added to the *original* stage cost for economic NMPC:

$$\phi(z_l, v_l) := \psi(z_l, v_l) + 1/2 \|z_l - z^*, v_l - v^*\|_Q^2 \quad (39)$$

For suitably large Q the corresponding $L(\bar{z}, \bar{v})$ becomes strongly convex and ensures that Assumption 2 holds.

To find valid diagonal regularization matrices Q one can apply the Gershgorin property for a matrix $A = \nabla^2 L(\bar{z}, \bar{v})$ with elements $a_{i,j}$, which states that

$$a_{i,i} - \sum_{j \neq i} |a_{i,j}| \leq \sigma_i \leq a_{i,i} + \sum_{j \neq i} |a_{i,j}|. \quad (40)$$

where σ_i are the eigenvalues of A . To ensure a regularization where the rotated

stage cost is strongly convex (and $A + Q$ has positive eigenvalues), we denote the diagonal elements of Q as q_i and require

$$0 < q_i + a_{i,i} - \sum_{i \neq j} |a_{i,j}| \leq \sigma_i + q_i, \quad (41)$$

i.e., $q_i > \sum_{i \neq j} |a_{i,j}| - a_{i,i}$. Nevertheless (41) must be satisfied *for all* z, v satisfying $z \in \mathbb{X}, v \in \mathbb{U}$, so some off-line pre-computation is needed. This condition determines a “minimal” regularization that guarantees stability of Economic NMPC.

Nominal stability of Economic NMPC with regularization based on strong convexity is analyzed in [25]. Moreover, ISS stability properties of regularized Economic NMPC formulations are also analyzed in [22], where this formulation is also extended to dynamic processes that operate in a cyclic steady state over a period of K sampling times. (The particular case with $K = 1$ is the normal steady state). Such systems arise under periodic variation of input conditions such as feedstocks, prices and boundary conditions. Cyclic systems, which are never in steady state, include periodic adsorption processes, reactors with reaction/regeneration cycles and processes which follow diurnal variations in energy prices.

Within this optimization framework, Economic NMPC has been demonstrated on a number of process applications. Two literature processes are considered in [2] and an extensive set of experiments compare performance and costs between tracking-type and Economic NMPC. These studies show that optimizing process economics takes advantage of the type and frequency of disturbances and can lead to an economic advantage of 5-10% in steady-state profit over tracking-type controllers.

For the dynamic real-time optimization of low density polyethylene (LDPE) reactors, significant production improvements can be made during fouling and defouling cycles [59]. In particular, optimal adjustment of the temperature profile can increase production levels by more than 10% compared to setpoint tracking.

For the control of compressors in gas pipeline applications, a number of advantages can be seen in tracking diurnal variations in electricity prices and cost savings of 5% were observed when choosing real-time pricing over flat-rate schemes [18].

Finally, for thermo-mechanical pulping in the paper industry, the integration of economic objectives with an NMPC formulation leads to potential reductions in energy cost of about 12% for current process configurations [20], and up to 24% with the addition of multiple pulping stages [21].

5.1. Distillation Column Case Study for Economic NMPC

We illustrate stability concepts for Economic NMPC on a distillation column case study described in [31]. The model is composed of two distillation columns in sequence with a ternary component system. The three components are A, B and C, where A is the lightest component, B is the middle one and C is the heaviest component. B is the most valuable product. For our example the three components are benzene (A), toluene (B) and p-xylene (C). Each column has 41 trays including the total condenser and the partial reboiler; the feed enters column 1 at stage 21, and the bottom product from column 1 enters column 2 at stage 21. Component A leaves the first column from the top; the bottom flow of the first column becomes the feed of the second column; component B leaves the second column from the top and component C leaves the second column from the bottom. The molar holdups in the condenser drums and reboilers are assumed to be controlled by the distillate and bottom flow rates, respectively. Therefore manipulated variables are $u = [LT1, VB1, LT2, VB2]$ which are reflux and boilup for each column. The structure of the two columns in sequence is shown in Fig. 4. The columns are modeled with ideal thermodynamics, assuming constant relative

Figure 4: Structure of two distillation columns in sequence

volatility and vapor flow rate for all stages. The problem formulation is given by:

$$\begin{aligned}
\min_u \quad & J(u) = p_F F + p_V (VB1 + VB2) - p_A D1 - p_B D2 - p_C B2 \quad (42) \\
\text{s.t.} \quad & \text{Mass balance, Equilibrium} \\
& x_A \geq x_{A,min}, x_B \geq x_{B,min}, x_C \geq x_{C,min} \\
& 0 \leq LT1, LT2 \leq LT_{max}, 0 \leq VB1, VB2 \leq VB_{max}
\end{aligned}$$

where D and B are the distillate and bottom flow rate of each column; p_F, p_V, p_A, p_B, p_C are prices for the feed, vapor, component A, component B and component C; their values and additional details of the model could be found in [31].

The Hessian matrix of the Lagrange function of the above steady state problem is not positive definite with $\sigma_{min} = -1.414$. Therefore if the current objective function is applied by eNMPC, the optimum steady state cannot be reached, and the system is not Lyapunov stable. Instead, the objective function must be modified to include *regularized* economic stage costs. This ensures that the first and third items of Assumption 2 are satisfied. Moreover, through offline trials, we determined that $N = 20$ is a sufficiently long horizon so that the second item of Assumption 2 is satisfied as well.

In order to regularize the stage costs, we add quadratic terms to the objective function to make the Hessian matrix positive definite. The regularization terms have the form of (39) where Q could be calculated by (41) for all values over a grid of (z, v) over the entire domain. The objective function then becomes:

$$\begin{aligned}
\min_u J(u) = \quad & p_F F + p_V (VB1 + VB2) - p_A D1 - p_B D2 - p_C B2 \\
& + \frac{1}{2} \| z - z^*, v - v^* \|_Q^2 \quad (43)
\end{aligned}$$

Since the size of the steady state problem is large (979 variables, 975 constraints), and the Hessian matrix of the Lagrange function of the regularized steady state problem is not constant but changes with different variable values, Gershgorin weights could not be calculated directly. Therefore MATLAB was used along with AMPL to search for Gershgorin weights. To ensure that the rotated stage cost is strongly convex, the Hessian has to be positive definite with all possible values of variables. We divide the feasible value of each variable into 40 grid points and calculate value of each element of the Hessian matrix at each grid point. Finally,

we have the following inequality:

$$q_i \geq \sum_{i \neq j} \max_{z,v} |a_{i,j}| - \min_{z,v} a_{i,i} \quad (44)$$

To minimally regularize the economic stage costs we apply exact Gershgorin weights calculated by

$$q_i = \sum_{i \neq j} \max_{z,v} |a_{i,j}| - \min_{z,v} a_{i,i} \quad (45)$$

and show the effect of Gershgorin weights by comparing the state profiles generated without and with regularization, and without and with disturbances.

This case study considers two cases: no disturbance, and 5% disturbance; in the second case disturbances are introduced as additive noise. The disturbances are Gaussian noises with zero mean, and their standard deviations are 5% of the optimum steady state value of each state. For each case the objective function has three formulations: economic cost only, economic cost plus regularization terms calculated by Gershgorin Theorem, and tracking terms only. We start from a state that is not the optimum steady state and continue until $time = 100$. Here, each of the 100 NMPC problems has 136154 variables and 77754 constraints.

Fig.5 is generated without disturbance. From Fig.5(a) we could see that when the objective function is composed of only economic cost, as shown in blue, x_A , which is the composition of component A, drifts away from its optimum steady state, and x_C , the composition of component C, diverges. If the objective only includes tracking terms, as shown in green, x_A , x_B and x_C converge from the starting point to the optimum steady state very fast, and as Fig.5(b) shows, the corresponding control profiles converge to their set points very fast. If the objective includes both economic cost and regularization terms whose weights are calculated using the Gershgorin Theorem, x_A , x_B and x_C converge to the optimum steady state very fast as well. In Table 1 we compare the time-accumulated economic cost, which is the objective function in (42), from $time = 1$ to $time = 100$. Here we observe that without regularization terms, the state does not approach the optimum steady state, and the cost is the highest. With regularization terms the cost is reduced significantly.

Fig.6 is generated with 5% additive noise. It could be observed that without regularization terms, the system is not Lyapunov stable; while regularization terms are able to keep the state within a small region around the optimum steady state. Again, in terms of accumulated economic cost, the cost is reduced with

(a) State profiles

(b) Control profiles

Figure 5: No disturbance. The results without regularization terms are shown in blue; the results with regularization terms calculated with Gershgorin Theorem are shown in red; the results with only tracking terms are shown in green.

Table 1: Cost of NMPC with different noise levels

Cost with	No noise	5% noise
No regularization	2.476	-19.97
Gershgorin weights	-22.27	-22.03
Tracking	-22.27	22.03

regularization terms.

Hence, without regularization, the controller is not Lyapunov stable. However, by adding regularization terms to make the Hessian matrix of the Lagrange function of the steady state problem positive definite, the controller could be stabilized at the optimum steady state. Another observation worth mentioning is the significant improvement in CPU time. Without regularization, it takes around 6 CPU mins to solve each NLP, while with regularization it only takes about 55 CPU seconds to solve each NLP. So CPU time drops significantly with regularization terms.

6. Conclusions and Future Work

Advances in large-scale nonlinear programming solvers and sensitivity lead to formulation of nonlinear model-based estimation (MHE) and control (NMPC) tasks that require only negligible on-line computation. This allows us to expand the scope of control activities from mere setpoint regulation to trajectory tracking and dynamic real-time optimization. Moreover, it allows us to incorporate large-scale nonlinear process models into dynamic on-line optimization strategies.

This study reviews recent results related to *advanced step* NMPC and D-RTO. We summarize nominal and robust stability properties for advanced step NMPC and discuss the extension of this strategy to clipping of the control variable in the first interval, as well as background calculations that are performed over multiple sampling times. In addition, we extend the NMPC formulation to consider economic stage costs that lead directly to dynamic real-time optimization. All of these advances are illustrated with case studies that use large-scale, nonlinear first principle distillation models.

These advances also lead to a number of open questions. Related to robust stability of NMPC, state variable bounds and inequalities are handled by penalties such that a robust positive invariant (RPI) set covers the entire state space.

(a) State profiles

(b) Control profiles

Figure 6: 5% additive noise. The results without regularization terms are shown in blue; the results with regularization terms calculated with Gershgorin Theorem are shown in red; the results with only tracking terms are shown in green.

On the other hand, the influence of uncertainty needs to be quantified based on measures of the ISS Lyapunov function in order to evaluate the performance of the NMPC controllers. Along with amsNMPC, multi-step variants of MHE can also be developed along with deeper analysis of robust stability that extend the results of Theorems 6 and 7. In addition, recent results for Economic NMPC raise interesting questions on the realization of dynamic RTO, the opportunity costs of operating at steady state and the degree of regularization required. Finally, unknown inputs and model parameters have a strong influence on the steady state values (z^*, v^*) in the economic NMPC regularization. In future, we plan to handle these through on-line state and parameter estimation, where (z^*, v^*) are also updated for each NMPC problem. For this, new state and parameter estimates will be used for solution of the steady state optimization problem, in order to update the regularized stage costs.

Acknowledgments

Many thanks to our research colleagues Devin Griffith, Eranda Harinath, Rui Huang, Johannes Jäschke, Rodrigo Lopez Negrete, Bethany Nicholson, Sachin Patwardhan and Victor Zavala for very useful input and discussions. Funding from National Science Foundation, Department of Energy (NETL) and Center for Advanced Process Decision-Making (CAPD) at Carnegie Mellon is gratefully acknowledged.

7. References

- [1] S. Abrol and T. F. Edgar. A fast and versatile technique for constrained state estimation. *Journal of Process Control*, 21(3):343 – 350, 2011.
- [2] R. Amrit, J. B. Rawlings, and L. T. Biegler. Optimizing process economics online using model predictive control. *Comp. Chem. Engr.*, 58:334–343, 2013.
- [3] D. Angeli, R. Amrit, and J. Rawlings. On average performance and stability of economic model predictive control. *IEEE Trans. Auto. Cont.*, 57 (7):1615–1626, 2012.
- [4] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50:174–188, 2002.

- [5] L. T. Biegler. A survey on sensitivity-based nonlinear model predictive control. *IFAC Symposium on Dynamics and Control of Process Systems (DYCOPS)*, 10:499–510, 2013.
- [6] L. T. Biegler and V. M. Zavala. Large-scale nonlinear programming using ipopt: An integrating framework for enterprise-wide dynamic optimization. *Computers and Chemical Engineering*, 33:575–582, 2009.
- [7] A. E. Bryson and Y. C. Ho. *Applied Optimal Control: Optimization, Estimation, and Control*. Taylor and Francis, New York, USA, 1975.
- [8] H. Chen and F. Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34:1205–1217, 1998.
- [9] F. Daum. Nonlinear filters: beyond the kalman filter. *Aerospace and Electronic Systems Magazine, IEEE*, 20(8):57–69, 2005.
- [10] M. Diehl, R. Amrit, and J. B. Rawlings. A lyapunov function for economic optimizing model predictive control. *IEEE Transactions on Automatic Control*, 56(3):703–707, 2011.
- [11] M. Diehl, H. G. Bock, and J. P. Schlöder. A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM J. Control and Optimization*, 43:1714–1736, 2005.
- [12] I.S Duff. MA57 - a code for the solution of sparse symmetric definite and indefinite systems. *ACM Transactions on Mathematical Software*, 30:118–144, 2004.
- [13] S. Engell. Feedback control for optimal process operation. *J. Proc. Cont.*, 17:203–219, 2007.
- [14] G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *Journal of Geophysical Research*, 99(C5):143–162, 1994.
- [15] A.V. Fiacco. *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*. Academic Press, New York, 1983.

- [16] G. A. G. Fischer and L. T. Biegler. Fast online optimization applied to high purity propylene distillation. Technical report, Carnegie Mellon University and Petroleo Brasileiro S.A, 2012.
- [17] A. Forsgren, P.E. Gill, and M.H. Wright. Interior methods for nonlinear optimization. *SIAM Review*, 44/4:525–597, 2002.
- [18] A. Gopalakrishnan and L.T. Biegler. Economic nonlinear model predictive control for the periodic optimal operation of gas pipeline networks. *Computers and Chemical Engineering*, 52:90–99, 2013.
- [19] G. Grimm, M. J. Messina, S.E. Tuna, and A.R. Teel. Examples when nonlinear model predictive control is nonrobust. *Automatica*, 40:523–533, 2004.
- [20] E. Harinath, L. T. Biegler, and G. A. Dumont. Control and optimization strategies for thermo-mechanical pulping processes: Nonlinear model predictive control. *Journal of Process Control*, 21:519 – 528, 2011.
- [21] E. Harinath, L.T. Biegler, and G. A. Dumont. Predictive optimal control for thermo-mechanical pulping processes with multi-stage low consistency refining. *Journal of Process Control*, 23:1001 – 1011, 2013.
- [22] R. Huang, L. T. Biegler, and E. Harinath. Robust stability of economically oriented infinite horizon NMPC that include cyclic processes. *J. Process Control*, 22 (1):51–59, 2012.
- [23] R. Huang, S. Patwardhan, and L. Biegler. Robust stability of nonlinear model predictive control with extended kalman filter and target setting. *International Journal of Robust and Nonlinear Control*, 23(11):1240–1264, 2013.
- [24] R. Huang, S. C. Patwardhan, and L. T. Biegler. Robust stability of nonlinear model predictive control based on extended kalman filter. *Journal of Process Control*, 22:82–89, 2012.
- [25] J. Jäschke, X. Yang, and L. T. Biegler. Fast economic model predictive control based on nlp-sensitivities. *Journal of Process Control*, 24:1260–1272, 2014.
- [26] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. Dover Publications, Inc., Mineola, New York, 2007.

- [27] Z.P. Jiang and Y. Wang. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37:857–869, 2001.
- [28] S. Julier, J. Uhlmann, and H.F. Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482, 2000.
- [29] S.S. Keerthi and E.G. Gilbert. Optimal infinite-horizon feedback laws for general class of constrained discrete-time systems: Stability and moving-horizon approximations. *IEEE Trans. Auto. Cont.*, 57:265–293, 1988.
- [30] M. Lazar, W. P. Maurice, H. Heemels, and A. R. Teel. Further input-to-state stability subtleties for discrete-time systems. *IEEE Trans. Auto. Cont.*, 58:1609–1613, 2013.
- [31] R. B. Leer. Self-optimizing control structures for active constraint regions of a sequence of distillation columns. Master’s thesis, Norwegian University of Science and Technology, 2012. Url: <http://www.diva-portal.org/smash/get/diva2:629241/FULLTEXT01.pdf>.
- [32] W. Li and L. T. Biegler. Multistep, newton-type control strategies for constrained, nonlinear processes. *Chem. Eng. Res. Des.*, 67:562–577, 1989.
- [33] R. López-Negrete and L. T. Biegler. A moving horizon estimator for processes with multi-rate measurements: A nonlinear programming sensitivity approach. *Journal of Process Control*, 22(4):677 – 688, 2012.
- [34] R. López-Negrete, F. J. D’Amato, L. T. Biegler, and A. Kumar. Fast nonlinear model predictive control: Formulation and industrial process applications. *Computers & Chemical Engineering*, 51:55 – 64, 2013.
- [35] L. Magni and R. Scattolini. Robustness and robust design of mpc for nonlinear discrete-time systems. In R. Findeisen, F. Allgöwer, and L.T Biegler, editors, *Assessment and Future Directions of Nonlinear Model Predictive Control*, pages 239–254. Springer, Berlin, 2007.
- [36] D.Q Mayne, J.R. Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: stability and optimality. *Automatica*, 36:789–814, 2000.

- [37] B. L. Nicholson, R. Lopez-Negrete, and L. T. Biegler. On-line state estimation of nonlinear dynamic systems with gross errors. *in press*, 2014.
- [38] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, USA, second edition, 2006.
- [39] T. Ohtsuka. A continuation/GMRES method for fast computation of nonlinear receding horizon control. *Automatica*, 40 (4):563 – 574, 2004.
- [40] G. Pannocchia, J.B. Rawlings, and S.J. Wright. Conditions under which suboptimal nonlinear mpc is inherently robust. *Systems & Control Letters*, 60:747–755, 2011.
- [41] B. Picasso, D. Desiderio, and R. Scattolini. Robust stability analysis of nonlinear discrete-time systems with application to mpc. *IEEE Trans. Auto. Cont*, 57:185–191, 2012.
- [42] J. Prakash, S. C. Patwardhan, and S. L. Shah. Constrained nonlinear state estimation using ensemble kalman filters. *Industrial & Engineering Chemistry Research*, 49(5):2242–2253, 2010.
- [43] J. Prakash, S. Shah, and S. Patwardhan. Constrained state estimation using particle filters. In M. J. Chung and P. Misra, editors, *Proceedings of the 17th IFAC World Congress*, volume 17, pages 6472–6477, 2008.
- [44] D. Ralph and S. Dempe. Directional derivatives of the solution of a parametric nonlinear program. *Mathematical Programming*, 70(1-3):159–172, 1995.
- [45] C. V. Rao, J. B. Rawlings, and D. Q. Mayne. Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *IEEE Trans. Auto. Cont.*, 48(2):246–258, 2003.
- [46] J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, LLC., 2009.
- [47] J.B. Rawlings and R. Amrit. Optimizing process economic performance using model predictive control. In L. Magni, D. M. Raimondo, and F. Allgöwer, editors, *Assessment and Future Directions of Nonlinear Model Predictive Control*, pages 119–138. Springer, 2009.

- [48] J.B. Rawlings, D. Bonné, J. Jørgensen, A.N. Venkat, and S.B. Jørgensen. Unreachable setpoint in model predictive control. *IEEE Trans. on Auto. Cont.*, 53:2209–2215, 2008.
- [49] M. Tenny and J.B. Rawlings. Efficient moving horizon estimation and non-linear model predictive control. In *Proceedings of the American control conference, Anchorage, AK.*, 2002.
- [50] P. Vachhani, S. Narasimhan, and R. Rengaswamy. Robust and Reliable Estimation via Unscented Recursive Nonlinear Dynamic Data Reconciliation. *J. Process Control*, 16:1075–1086, 2006.
- [51] P. Vachhani, R. Rengaswamy, V. Gangwal, and S. Narasimhan. Recursive estimation in constrained nonlinear dynamical systems. *AIChE Journal*, 51(3):946–959, 2004.
- [52] A. Wächter and L. T. Biegler. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.
- [53] I. Wolf, L. Würth, and W. Marquardt. Rigorous solution vs. fast update: Acceptable computational delay in NMPC. In *Decision and Control and European Control Conference, Orlando, Florida, USA*, pages 5230–5235. 2011.
- [54] L. Würth, R. Hannemann, and W. Marquardt. Neighboring-extremal updates for nonlinear model-predictive control and dynamic real-time optimization. *J. Process Control*, 19:1277–1288, 2009.
- [55] L. Würth, J. B. Rawlings, and W. Marquardt. Economic dynamic real-time optimization and nonlinear model predictive control on infinite horizons. In *International Symposium on Advanced Control of Chemical Process*, Istanbul, Turkey, 2009.
- [56] X. Yang and L. T. Biegler. Advanced-multi-step nonlinear model predictive control. *J. Process Control*, 23:1116–1128, 2013.
- [57] V. Zavala and L. Biegler. The advanced step nmpc controller: Optimality, stability and robustness. *Automatica*, 45:86–93, 2009.

- [58] V. M. Zavala and L. T. Biegler. Nonlinear programming strategies for state estimation and model predictive control. In F. Allögger (Eds.) L. Magni, D. Raimondo, editor, *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, pages 419–432. Springer, Berlin, 2000.
- [59] V. M. Zavala and L.T. Biegler. Optimization-based strategies for the operation of low-density polyethylene tubular reactors: Nonlinear model predictive control. *Computers and Chemical Engineering*, 33:1735–1746, 2009.
- [60] V. M. Zavala, C. D. Laird, and L. T. Biegler. A fast moving horizon estimation algorithm based on nonlinear programming sensitivity. *Journal of Process Control*, 18(9):876–884, 2008.